

FIG. 1

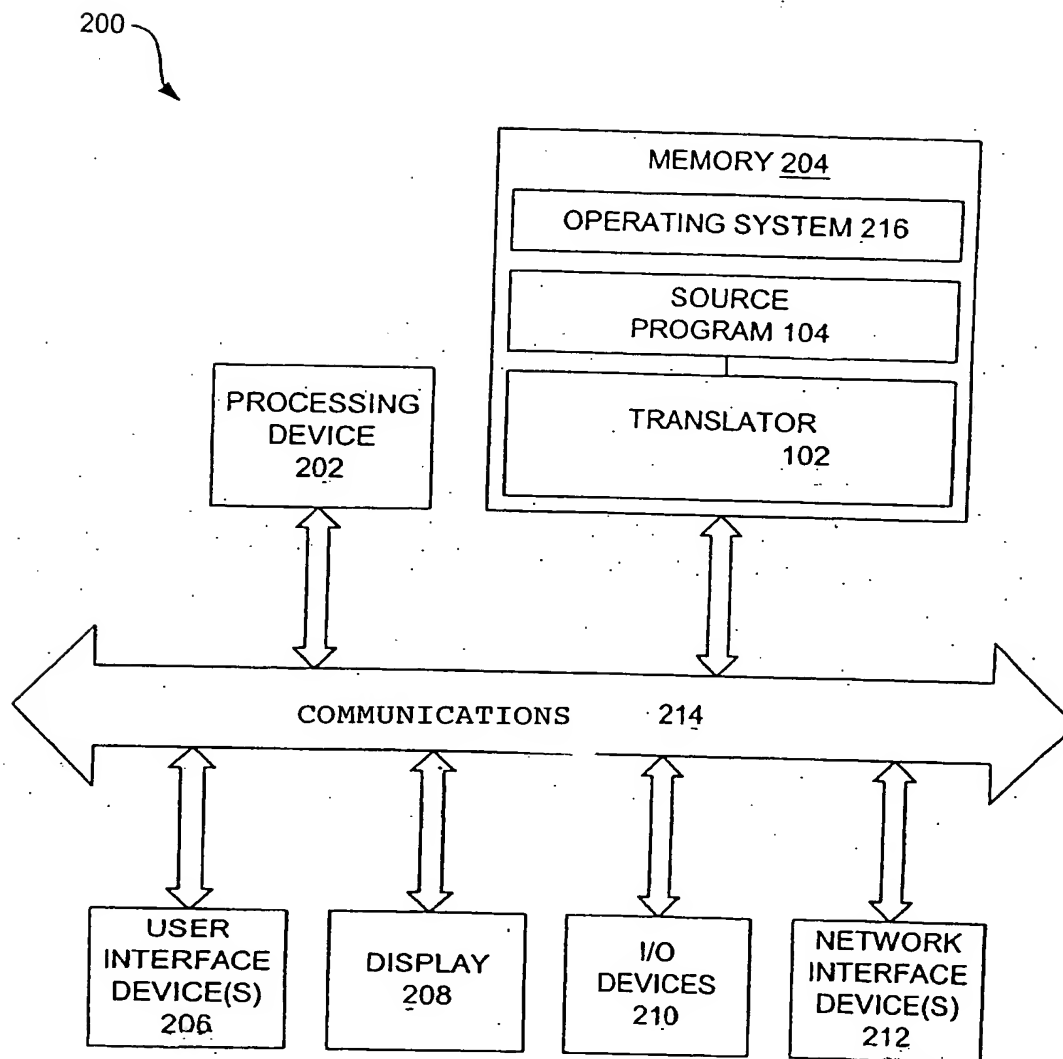


FIG. 2

300

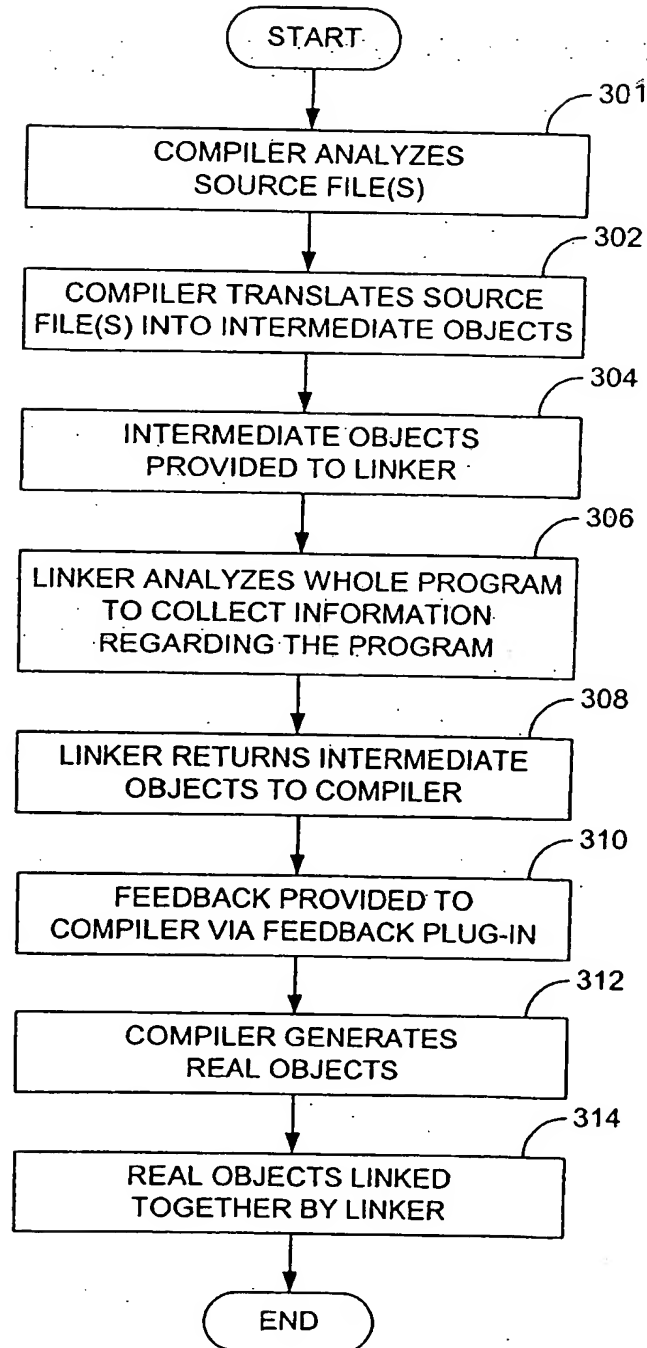


FIG. 3

```
int g;

void bar(int** fp1, int** fp2, int i)
{
    *fp1 = &g;
    *fp2 = (int*) malloc(i*sizeof(int));
    return;
}

int foo()
{
    int* fp1, *fp2, i;

    bar(&fp1, &fp2, 100);
    g = 1;

    fp2[0] = 10;

    for (i = 1; i < 100; i++)
    {
        fp2[i] = 2*g + 10;
    }

    *fp2+= 10;          (1)

    *fp1 = 20;          (2)

    return g;
}
```

**FIG. 4**

```
int foo()
{
    int *fp1, *fp2, i;
    bar(&fp1, &fp2, 100);

    fp2[0] = 10;

    for (i = 1; i < 100; i++)
    {
        fp2[i] = 12;
    }
    *fp2 += 10;
    *fp1 = 20;
    return 20;
}
```

**FIG. 5**

```
int * gp;

int foo(int *f)
{
    int *l = 0;

    l = f;    (1)    // <- this assignment need not be
represented in SIR

    ....

    gp = l;    (2)    // <-- this assignment to be
represented as gp = f in SIR.

}
```

**FIG. 6**

```
struct A {  
    int i;  
    int* fd;  
    ...  
};  
  
extern int g[]  
extern int* gp;  
  
int foo(A* f, int f2)  
{  
    A* lp;  
    ....  
    if (f2 > 10)  
        lp = f->fd;  
    else  
        lp = (char*)&g + 12;  
  
    gp = lp;    (1)  
    ..  
}
```

**FIG. 7**

```
int* gp;

int foo(int i)
{
    int* t = (int*) malloc(i); (1)

    ...
    gp = t; (2)
    bar(); (3)
    ..
}
```

**FIG. 8**